

FOUNDATIONS AND SYSTEMS CG&A article

P.Robertson, R.Earnshaw, D.Thalman, M.Grave, J.Gallop, E. De Jong

Many visualization tools and techniques used by scientists are integrated to some degree within a system. Few systems, however, fully meet their users' needs. Limited functionality, limited information about implicit assumptions or embedded constraints, and incomplete integration of different tools, are all common problems experienced. These problems are exacerbated because visualization incorporates not just the graphics tools, but also the analysis and presentation of data, exploration of data and problems, and interaction with models and simulations.

Partly these limitations arise from the historical evolution of proprietary or application-specific systems. But they also arise from the lack of a clear articulation of the foundation assumptions and practices of scientific visualization. The application of these assumptions and practices in building tools in a systematic manner, and in clarifying and validating visualizations, also requires articulation.

We focus here on the research required to provide a sufficient foundation to meet the evolving needs of future visualization systems, as determined during the workshop discussions. We believe that three main topics underpin these needs:

- * Models: the need for abstractions to describe the core components or elements of the process of visualization, and the interfaces between these components, including users and their behaviour.
- * Validation: the problem of determining whether visualizations, generated according to prescribed design criteria or realised using standard or prescribed tools and techniques, meet consistency and effectiveness criteria on test data or measures.
- * Systems: the design, realisation and operational problems of visualization systems that integrate a range of functionalities to provide a working environment for scientists using visualization tools.

We treat the need for models with the highest priority. This is because the systems in widespread use have reached the point where it is difficult to make further progress without addressing more formally the underlying "models" they encapsulate either implicitly or explicitly. Flexible and evolving models have a potential to clarify requirements for validation, and for the design of systems, within a unified framework.

In the following sections we outline key aspects of each area, commenting on the current status of work, and isolating areas where significant research effort is still required. We also suggest possible strategies for initiating this research.

MODELS

There is a clear need not only for an overall reference model of the visualization process, but also of models for key components that can

be used independently if required. We consider a data model as absolutely essential if new application software is to be written in a manner that allows its use across different environments. We also believe that time has not been treated sufficiently formally to track independently, and align as required, for example, simulation time, computation time, simulated timeframe, video recording time, playback time, and various events that may control, or arise from, modelling and visualization. A third important area is a user model, to allow clear indication of expectations, assumptions, and implications relating to specific and general users interacting with visualization systems. In particular, expectations for decoding of visualizations are difficult to state objectively without reference to a user model. In this section we summarise key aspects of each of these three areas, in addition to an overall reference model.

Reference Model

Several workshops have raised the need for a reference model for the whole visualization process. A reference model, formally described, can separate clearly components of the process by identifying core functionalities, and can serve as a basis for standardising terminology, comparing systems, finding choice of systems, and focussing on constraints or limitation of our current understanding of the process. There is currently no such widely recognised and accepted reference model for visualization to meet these requirements, although there have been partial models proposed. As a consequence, terminology varies and comparisons are difficult. The existing de-facto model - the data flow pipeline - does not fully describe even the systems in use currently and their integration with modelling of simulation. There is currently no recognised formal description of even this data flow model and its components. There is thus a strong need for an initial reference model, and defined terminology, that is sufficiently flexible to evolve as the field matures.

Data Models

Despite widely available tools to convert between the large range of data formats used in visualization, and despite many calls for the development of data models to support application development, the community has not established data models that describe adequately the full range of data used - it is not enough to say "this data is 5-D". Some attempts to classify data types and characteristics have been made [Haber91, Brodlie92]. Data models that can fully describe the data at a generic level are required if applications are to be able to handle and process data at increasingly semantic levels. A data model could also clarify to a user or application whether, and why, processes may be applicable to some data but not to other data.

The ability to interact at a level of abstraction from the generic structure of the data, while maintaining the full integrity of the data structures and descriptive information, is critical if applications developers are to be freed from low-level domain-independent tasks and allowed to concentrate on high-level domain-specific tasks. Achieving this will help us to describe visualization systems more formally, and underpin the development of visualization reference models. The recent proposals and experiments in automatic systems for visualization also rely on the data being described adequately. Object-oriented design methodologies currently offer the most promising framework for the development of data models, but the difficulty of the problem should not be underestimated.

User Models

In visualization, there is always reference to user's needs, and much work is devoted to specifying who these users are, and what kinds of needs they have [Upson89]. There are at least four criteria for differentiating users.

First, some users are what is usually called "end-users", which means that they are scientists, having no knowledge of computer science or programming, who use computers and associated software as tools, without any knowledge of the way they have been built (like a text processor or a video game for example). There are also users who develop programs, and use libraries. The needs of these users are not the same, and a clear classification can be made based on their different involvement in computing.

A second way of classifying users is according to the application domain they are working in. Needs in biology are different from those in pure mathematics, or in fluid dynamics.

A third method for differentiating users can be based on the types of visualization tools they need. In some cases, large numbers of 2D curve plots are needed, for example for measuring with a rule, while in other cases pictures of 3D objects rendered with advanced tools are expected, for example for making a film to be presented in a conference. A single user may also have these two different types of needs at different stages of work.

Finally, it is important to specify the way users want to operate their visualization tools. Some post-process their results in batch mode, some look at the results while they are produced by a simulation or an experiment, and some even use visualization tools for influencing the behaviour of a simulation or an experiment.

It is important to be able to describe precisely these factors that characterize a user's behaviour in order to understand how well visualization tools correspond to what is expected, and to be able to assess effectiveness.

Time Model

Most of the phenomena which may be represented on the screen of a workstation are typically time-dependent. In order to visualize these phenomena at any given time, it is necessary to know the appearance of the scene at this time. Computer Graphics techniques then allow us to build and display the scene according to viewing and rendering parameters. We have to be able to express time dependence in the scene, and how to make the scene evolve over time. These problems and their various solutions are part of what is usually understood by the term Computer Animation.

Time cannot be treated in the same way as a spatial variable is treated for several reasons:

- time is always positive
- many processes are not only time-dependent, but also time-critical
- there are different kinds of time to consider.

A time model is required in order to formally describe the time variable and its relation to all processes involving it.

This model should take into account the following aspects of time:

- simulation time
- simulated timeframe
- computation time
- recording time (e.g. using video)
- playback time
- user's timeframe.

The time model should clearly take into account the time characteristics of the process to be visualized and the time characteristics of the animation method.

We also think it very important to formally define models of synchronization of time-dependent processes, including full treatment of events.

Device models

Visualization processes need to enter large amounts of data into the computer, to interact with them, to decide the kind of visualization procedure to use, and to choose relevant parameters. All these actions require the use of devices. As the classical input devices (keyboard and mouse) are progressively replaced by 3D devices, VR devices and multimedia devices, there is a clear need for input device models. There is also a need for well-defined output device models, and device models that describe the integration of input and output functionalities, with any associated constraints, to allow interaction. These models should at the very least identify devices in terms of the nature of data they may accept, and their functionalities in relation to that data.

For example, data characteristics include:

- geometric: 3D positions, 3D orientations, trajectories, shapes, deformations
- kinematics: velocities, accelerations, gestures
- dynamics: forces and torques in physics-based animation
- lights and colors - sounds - commands

Integrated input/output device models should incorporate:

- 3D position/orientation measurement device (e.g. DataGlove, 3D mouse, SpaceBall)
- multidata input devices (e.g. MIDI keyboard)
- head-mounted displays (e.g. EyePhone)
- audio input
- audio output
- video input (e.g. VideoLab, videodisk)
- video output (e.g. VideoLab, videodisk)

Device models should also take into account motion capture models and methods such as:

- recording input data from a device in real-time while simultaneously applying the same data to a graphics object on the screen.
- recording input data from a device in real-time and producing effects of a different nature but corresponding to the input data.

- recording input data from a device in real-time and analyzing them.

VALIDATION AND CERTIFICATION

Currently there are few, if any, tests or checks to ensure that the results of using visualization systems are correct. It is therefore highly desirable that standardized data sets be designed and made available that can act as benchmarks by which visualization systems can be quantitatively and objectively measured and assessed. This will also enable new algorithms and new systems to be tested before they are released for general use. Unless users put pressure on designers, implementors and vendors to perform such validation tests, they will not happen.

Effectiveness Metrics

Measures of the effectiveness of the presentation of visualizations are needed in order to assist with:

- determining the advantages and disadvantages of the different presentation methods available
- selection of the most appropriate methods for a specific application or discipline
- selection of the most appropriate methods for application-independent tasks such as locating scale-dependent anomalies in data
- selection of appropriate colours, or other display parameters, to minimise the risk of misleading users.

Further input and work from psychologists and perception experts is needed in this area.

SYSTEMS

There is a wide range of systems, commercial, public-domain, and "in-house", currently used for scientific visualization. Most of the current systems have one or more of the following limitations:

- large data sets are cumbersome to handle
- large programs are cumbersome to handle
- large module suites are unmanageable
- systems are not optimised, so run slowly
- user extension of the system needs expertise
- use of distributed computing needs care
- visual programming style is not natural for all users

Most of the commonly used software systems work according to the 'dataflow paradigm'. That is to say, data is read in at one end and then subjected to various transformations (selected from the modules available) and then rendered ready for display on the workstation. Any change to the data set must be read in once more and the process repeated. Thus interaction rates are dependent principally on the speed with which the data can be processed down the dataflow pipeline. For large data sets (e.g. 500 by 500 by 500) there are currently severe limitations on the speed with which one interacts with the data caused simply by the time taken to process the data.

Integration of computation with the visualization is an increasingly important requirement where users require to steer the computation in areas of particular interest. This approach requires the interface to the data model to be re-examined and a more efficient access method produced.

Default parameters

Many systems offer similar implementations of published algorithms, but few systems generate identical results for a given data set. We believe that standard algorithm parameters should be given recognised defaults to allow users to compare results, and where required, generate identical results across different packages. This requires identification of key algorithms or processes, recommendation of a set of default parameters related to standard test data sets, and participation from system vendors to provide implementation of agreed default settings within their systems.

Data handling / databases

Applications are increasingly needing database functionalities to satisfy integrity, historical trace, and semantic query requirements. Databases generally do not provide the performance required for large data sets, and it is not clear that data structuring is currently sufficiently explicit in most visualization environments to allow the full capabilities of databases to be exploited. It is even less clear whether visualization practitioners have actually made serious efforts to establish whether databases could meet some of their needs for data handling.

The problems of handling large data sets across networks, or distributed storage, while meeting interaction time requirements, is becoming more evident as video is increasingly used to communicate. Data tiling techniques are evolving concise descriptive languages, but these are not in general integrated with databases, or even accessible within system building tool-kits.

There is thus a clear need for a coordinated approach to integrating the handling of large data sets within databases for visualization environments. Several special sessions or workshops are emerging to tackle this problem.

Interoperability

Visualization tools cover a large range of complexity, from the simple curve plotter, to the large assembly of complex tools, including for example advanced 3D renderers, or even mesh generators. In many cases, the same user (or group of users) needs to use several of them, on the same data, or in the same environment, at different stages of work, or because of a need to perform different processing on them.

Since it is illusory to think about having a unique universal system capable to handle all types of problems, one of the important issues in scientific visualization is the definition of ways to allow different tools to be used together. A user should therefore be able to export parts of the data from one tool to another. This could be achieved:

- through a socket-type explicit connexion between tools.

- by generation of data in a file or a data-base management system.
- by the use of a specific mechanism in the execution environment, acting like for example the "Clipboard", in the MacIntosh environment.

This does not only mean that standard ways of coding and storing data have to be defined, but also that communication mechanisms between tools have to be define. Therefore more global work on environments in which visualization tools are used is required.

Distributed Systems

Today's computing environments are distributed and heterogeneous because all computers do not have the same properties. Sharing of resources is now easier than in the past. The concept of a server (for files, graphics or number crunching) is widely used in visualization, and implementation of distributed tools is now required. Moreover, the increasing number of parallel systems is an other important reason for building applications in terms of cooperating tasks.

The generality of UNIX as an operating system in the scientific community facilitates a lot the implementation of such systems, but there is still a lot of work to do at a higher level, especially in shared data management.

Cooperative working

Scientific work, among others, require more and more cooperation between people not necessary at the same location. The large availability of networks, and the impressive progress in user interfaces and multimedia technologies can provide good tools for "Computer Supported Cooperative Work".

The next generation of Visualization tools will have to provide such facilities, and allow for example two scientists at different sites to analyze jointly the results of a computer simulation, and interact with each other.

Automated approaches

To move from ad-hoc to principled visualization design, that does not rely on visualization experts to guide choice of visualization, next generation systems must incorporate guidance based on data characteristics and interpretation aims, and the available representation. Approaches to incorporating knowledge, systematic choosing of representations, and identifying interpretation aims, are in early stages of development. Data structures also do not currently carry sufficient information to allow such guidance to be applied effectively. Research is needed into levels of guidance required, interfaces to provide access to such guidance, and approaches to constructively composite displays.

Multimodal interfaces

The traditional main difficulty in the process of interacting with visualization systems is the lack of 3D interaction. Visual feedback, in a typical computer graphics application that requires items to be positioned or moved in 3-D space, usually consists of a few orthogonal and perspective projection views of the same object in a multiple window format. This window layout creates a virtually unsolvable puzzle for the brain and makes it very difficult (if not impossible)

for the user of such interfaces to fully understand his work and to decide where further alterations should be made. Moreover good feedback of the motion is almost impossible making the evaluation of the motion quality very difficult. Today, new technologies may immerse us in these computer-generated worlds or at least communicate with them using specific devices. In particular, with the existence of graphics workstations able to display complex scenes at interactive speed, and with the advent of such new 3-D interactive devices as the SpaceBall, EyePhone, and DataGlove, it is possible to create applications based on a full 3-D interaction metaphor in which the specifications of visualization and motion are given in real-time. This new concepts drastically change the way of interacting with visualization systems.

However, in order to take fully advantages of new technologies. user interfaces should not be based only on 3-D graphics but become truly multimodal interfaces. We should consider multimedia devices like MIDI keyboards, real-time video input/output devices and even audio input/output devices and also force-feedback devices. Gesture-based metaphors based on neural networks should be also investigated as part of a multimodal interface.

Hardcopy

Hard-copy media, including paper, transparencies and video are, and will continue to be, important for communicating scientific insights and results, for decision support, and for large format displays such as maps, where overlay of other hardcopy material is often required.

Video can be critical for the study of time-dependent phenomena, or for exploiting the human visual system's temporal capabilities.

One problem is the difficulty of obtaining the colours that the user intends. Although the difference between printer and display gamuts mean that some compromises are necessary, significant colour faults on the printed image (a colour cast leads to greys that are not grey; loss of distinguishability between colours) can be reduced by careful colour matching. Software to help users match colours correctly (whether from a printer manufacturer or in the form of generic software) is needed, as is the more widespread use of perceptually linear colour models for communicating with printers.

Video also has pitfalls for those inexperienced with the medium. It is becoming much easier to send images to a videotape, either by sending images and a script to a central high quality video facility, or increasingly by directly connecting video equipment to a workstation. The medium is nevertheless difficult to use effectively.

Technological factors, such as controlling colours and line thickness over varying scale changes, pose problems that are seldom handled transparently to a user. It can be difficult for the video-novice to convey the message effectively, making good use of annotation and motion. For greatest effectiveness, scientists really need to incorporate the type of design expertise found in film and television studios. They also need better tools to provide control over such technological aspects as aliasing temporal filtering, and colour gamut mapping.

CONCLUSIONS

Strategy for making progress on developing improved models

How do we ensure that research on developing models for visualization is undertaken with the imperative we feel is required? Several previous visualisation workshops have highlighted the need for reference models, and an increasingly formal approach, but few have emerged. And we have not seen a comparison of visualization reference models against existing graphics reference models, apart from some early pipeline oriented discussions.

Perhaps the most effective way of overcoming the reluctance to formalise models in an evolving field is to establish special sessions at a visualisation conference, or hold a dedicated workshop. We suggest that this could be done at, for example, the IEEE Visualization conference in 1994, and that initial models from which progress could be made could emerge in a 1-2 year time-frame. We might expect standardisation on a reference model, and components of such a model, including data, user, time and device models, in around 5 years.

Strategy for making progress on validation and certification of tools and systems

How do we ensure that validation receives enough attention to avoid the risk that as computational platforms offer the performance needed for visualisation, we are able to have confidence in the validity and effectiveness of the tools we develop and use? Maintaining such a credibility could be very important for the future support the field receives from funding and decision making bodies.

Clearly a disciplined effort is needed in establishing some test data sets and results, and benchmarking commercial software. The supercomputing community found it necessary to do this to maintain research and commercial credibility, and there is every reason to suggest that the visualization community will have to do the same. Industry and research consortia, perhaps through a dedicated workshop or through a major society such as ACM or IEEE, would be best placed to undertake reproducibility test design.

Research groups will need to determine how best to test the effectiveness of visualizations by establishing a major focus in this area, drawing from expertise in psychology and cognitive science. While there is some effort in this area, substantially more is required if we are to have any real faith in the effectiveness of the visualizations we produce. Empirical experiments and the development of systematic frameworks are both required.

Strategy for making progress on developing improved systems

How do we ensure that research on developing systems for visualization is undertaken to result in better integration of tools and techniques within usable environments?

We believe that there is a strong contribution to be made from computer scientists in applying modern software engineering approaches to many of these problems. There is a need for research into the design of systems that are sufficiently flexible to allow for minimal latency of interaction, for example, over a distributed computing

environment. Achieving interoperability, effective use of distributed systems, and progressively more automated generation of visualizations are significant research problems that will take some years before progress is realised within commercially available systems.

Some aspects of the limitations of systems, however, could be addressed by consortia of research and industry groups: establishing default parameters, making available database interfaces, standardizing on multimodal device interfaces, and standardizing on device-independent color coordinate systems, for example, could make current systems more usable within a 1-3 year time-frame.

We have summarised the key research issues that evolved from the workshop discussions and pointed to some steps that could be taken to address these issues. Our coverage is by no means exhaustive, and should not be taken to suggest that there is no research being done in the areas we have highlighted. We recognise that important steps are currently being taken, and in many cases these works have drawn attention to opportunities. Almost all the issues that we have treated specifically arose in several of the workshop working groups, reinforcing our belief that they are fundamental to the development of visualization systems that are increasingly usable to the scientific community.

References

- [BERG89] R. D. Bergeron & G. G. Grinstein. A Reference Model for the
 □ Visualisation of Multi-Dimensional Data, Proc. Eurographics'89,
 □ Hamburg, FRG, 4-8 Sept, 1989 (Hansmann, W., Hopgood, F.R.A., &
 □ Strasser, W. eds), pp.393-399.
- [BROD92] K.W.Brodie et al, "Scientific Visualization: Techniques
 □ and Application", Springer-Verlag, 1992
- [EARN93] R.A.Earnshaw, M.A.Gigante and H.Jones (eds), Virtual Reality Systems,
 Academic Press, 1993
- [DYER90] D. S. Dyer. A dataflow toolkit for visualization. IEEE Computer
 □ Graphics and Applications, 10(4),: 60-69, July 1990.
- [HABE90] R. B. Haber & D. McNabb. Visualization idioms: A conceptual model
 □ for scientific visualization systems, 74-92. In G. Neilson &
 □ B. Shriver, eds, Visualisation in Scientific Computing, IEEE
 □ Computer Society Press, 1990.
- [HABE91] R. B. Haber, B. Lucas & N. Collins. A Data Model for Scientific
 □ Visualisation with Provisions for Regular and Irregular Grids.
 □ Proc. Visualisation'91, San Diego, 298-305, Oct 1991.
- [MAGN90] N. Magnenat Thalmann and D. Thalmann, Computer Animation: Theory
 and Practice, 2nd edition, Springer-Verlag, Tokyo, 1990
- [MAGN93] N. Magnenat Thalmann and D. Thalmann (eds), Virtual Worlds and
 Multimedia, John Wiley and Sons, 1993
- [MAGN93b] N. Magnenat Thalmann and D. Thalmann, "Computer Animation: A Key
 Issue for Time Visualization", Proc. ONR Workshop on Data

Visualization

- [ROBE91] P. Robertson. A methodology for choosing data representations.
[] IEEE Computer Graphics and Applications, 11(3): 56-67, May 1991.
- [THAL93] D. Thalmann, "Using Virtual Reality Techniques in the Animation Process in [EARN93], pp.143-159.
- [TREI90] L. Treinish. SIGGRAPH '90 Workshop Report: Data structures and
[] access software for scientific visualization. Computer Graphics,
[] 25(2):104-118, April 1991.
- [UPS089] C. Upson et al. The application visualization system:
[] A computational environment for scientific visualization.
[] IEEE Computer Graphics and Applications, 9(4):30-42, July 1989.
- [] A Formal Specification of a Graphics System in the Framework of the
[] Computer Graphics reference Model. Computer Graphics Forum. 12(1).
[] March 1993. pp3-20.